

Remarks

I. 35 U.S.C. §102

A. The Standard

“Under 35 U.S.C. § 102, every limitation of a claim must identically appear in a single prior art reference for it to anticipate the claim.” *Gechter v. Davidson*, 116 F.3d 1454, 1457 (Fed. Cir. 1997). “Anticipation requires the presence in a single prior art reference disclosure of each and every element of the claimed invention, arranged as in the claim.” *Lindemann Maschinenfabrik GmbH v. American Hoist & Derrick Co.*, 730 F.2d 1452, 1458 (Fed.Cir. 1984). As stated in *Scripps Clinic & Research Foundation v. Genentech Inc.*, 927 F.2d 1565, 1576 (Fed. Cir. 1991), “anticipation requires that all of the elements and limitations of the claim are found within a single prior art reference. . . . There must be no difference between the claimed invention and the reference disclosure, as viewed by a person of ordinary skill in the field of the invention.”

B. The Office Action Assertions

Claims 1, 3, 4, 6-13, 15 and 16 stand rejected under 35 U.S.C. §102. The Office Action states:

Claims 1, 3-4, 6-13, 15-16 are rejected under 35 U.S.C. 102(e) as being anticipated by Muller et al. (hereinafter “Muller”, 6,650,640 B1).

As per claim 1, Muller discloses an interface device connectable to a network, a computer and a storage unit, the interface device comprising:

- A sequencer including a hardware logic circuit configured to process a transport layer header of a network packet (column 4, lines 48-50, column 7, lines 20-25, 31-35, 64-67, column 8, lines 1-5, 17-20, 50-60, column 9, lines 1-5, column 15, lines 35-38, column 35, lines 53-67, column 36, lines 11-30);
- A memory adapted to store control information regarding a network connection being handled by said device (column 4, lines 20-25, column 9, lines 14-16, 20-25, 56-58, column 10, lines 1-7, column 11, lines 46-59, column 12, lines 11-15, column 52, lines 64-67, column 53, lines 1-7);
- A mechanism for associating said packet with said control information and for selecting whether to process said packet by said computer or to send data from said packet to the storage unit, thereby avoiding the computer (column 4, lines 45-50, 58-67, column 8, lines 50-60, 66-67, column 9, lines 13-17, 22-35, 66-67,

column 10, lines 2-7, column 11, lines 46-59, column 12, lines 11-15, column 16, lines 59-67).

Applicants respectfully disagree with several of the foregoing Office Action statements. For instance, applicants respectfully assert that neither column 4, lines 45-50, 58-67, column 8, lines 50-60, 66-67, column 9, lines 13-17, 22-35, 66-67, column 10, lines 2-7, column 11, lines 46-59, column 12, lines 11-15, nor column 16, lines 59-67 of Muller discloses "A mechanism for associating said packet with said control information and for selecting whether to process said packet by said computer or to send data from said packet to the storage unit, thereby avoiding the computer." Instead, column 4, lines 45-50 and 58-67 of Muller state:

When a flow packet is received at the network interface, a flow database manager receives the packet's flow key. The flow key may be assembled by a header parser module that parses a header portion of the packet. The flow database manager may also receive control information concerning the packet, such as an indication of the size of a data portion

...
manager associates an operation code with the received packet to indicate how the packet may be further processed by the network interface and/or a host computer. The specific operation code assigned for a packet may indicate whether the packet contains data that can be re-assembled with other data passed in the flow, whether the packet is a control packet or is otherwise devoid of data, whether the packet should not be processed through a particular network interface function (e.g., due to a flag in a header of the packet), etc.

Similarly, column 8, lines 50-60 and 66-67 of Muller state:

Header parser 106 parses a header portion of the packet to retrieve various pieces of information that will be used to identify related packets (e.g., multiple packets from one same source entity for one destination entity) and that will affect subsequent processing of the packets. In the illustrated embodiment, header parser 106 communicates with flow database manager (FDBM) 108, which manages flow database (FDB) 110. In particular, header parser 106 submits a query to FDBM 108 to determine whether a valid communication flow (described below) exists between the source entity that sent a packet and the destination entity. The destination entity may comprise an application program, a communication module, or some other element of a host computer system that is to receive the packet.

...
one source entity to one destination entity. A flow may be identified by a flow key assembled from source and desti-

Likewise, column 9, lines 13-17, 22-35 and 66-67 of Muller state:

Thus, for purposes of flow management, header parser 106 passes the packet's flow key to flow database manager 108. The header parser may also provide the flow database manager with other information concerning the packet that was retrieved from the packet (e.g., length of the packet).

...
entity served by NIC 100. Thus, FDBM 108 updates FDB 110 as necessary, depending upon the information received from header parser 106. In addition, in this embodiment of the invention FDBM 108 associates an operation or action code with the received packet. An operation code may be used to identify whether a packet is part of a new or existing flow, whether the packet includes data or just control information, the amount of data within the packet, whether the packet data can be re-assembled with related data (e.g., other data in a datagram sent from the source entity to the destination entity), etc. FDBM 108 may use information retrieved from the packet and provided by header parser 106 to select an appropriate operation code. The packet's operation code is then passed back to the header parser, along with

...
Now the packet may be stored in packet queue 116, which holds packets for manipulation by DMA (Direct Memory...

In addition, column 10, lines 2-7 of Muller state:

addition to storing the packet in a packet queue, a corresponding entry for the packet is made in control queue 118 and information concerning the packet's flow may also be passed to dynamic packet batching module 122. Control queue 118 contains related control information for each packet in packet queue 116.

Similarly, column 11, lines 46-59 of Muller state:

In state 136, information extracted from the packet's headers is forwarded to flow database manager 108 and/or load distributor 112. The FDBM uses the information to set up a flow in flow database 110 if one does not already exist for this communication flow. If an entry already exists for the packet's flow, it may be updated to reflect the receipt of a new flow packet. Further, FDBM 108 generates an operation code to summarize one or more characteristics or conditions of the packet. The operation code may be used by other modules of NIC 100 to handle the packet in an appropriate manner, as described in subsequent sections. The operation code is returned to the header parser, along with an index (e.g., a flow number) of the packet's flow in the flow database.

Continuing, column 12, lines 11-15 of Muller state:

Also in state 140, control information for the packet is stored in control queue 118 and information concerning the packet's flow (e.g., flow number, flow key) may be provided to dynamic packet batching module 122.

...

In state 142, NIC 100 determines whether the packet is...

And finally, column 16, lines 59-67 of Muller state:

The information generated by the header parser includes, in particular, a flow key with which to identify the communication flow or communication connection that comprises the received packet. In one embodiment of the invention, data from packets having the same flow key may be identified and re-assembled to form a datagram. In addition, headers of packets having the same flow key may be processed collectively through their protocol stack (e.g., rather than serially).

None of the above-quoted passages that were cited by the Office Action teaches or suggests “a mechanism for associating said packet with said control information and for selecting whether to process said packet by said computer or to send data from said packet to the storage unit, thereby avoiding the computer,” as recited in claim 1. For at least this reason, claim 1 is not anticipated by Muller.

Regarding claim 3, the Office Action states:

As per claim 3, Muller discloses the interface device of claim 1, further comprising a plurality of network ports, wherein one of the said network ports is connectable to a storage unit (column 4, lines 40-43, column 6, lines 37-40, column 7, lines 15-19, column 8, lines 40-43, column 9, lines 1-5, column 10, lines 65-67).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 3 and respectfully assert that none of those passages teaches or suggests “The interface device of claim 1, further comprising a plurality of network ports, wherein one of said network ports is connectable to the storage unit.” For at least this reason, claim 3 is not anticipated by Muller.

Regarding claim 6, the Office Action states:

As per claim 6, Muller discloses the network interface device of claim 1, further comprising a file cache adapted to store said data (column 56, lines 20-30, column 58, lines 26-30, column 61, lines 34-35, column 62, lines 47-52).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 6. Without reprinting each of those passages here, suffice it to say that applicants respectfully assert that none of those passages teaches or suggests “The network interface device of claim 1, further comprising a plurality of network ports, wherein one of said network ports is connectable to the storage unit.” For at least this reason, claim 3 is not anticipated by Muller.

Regarding claim 4, the Office Action states:

As per claim 4, Muller discloses the interface device of claim 1, further comprising a Fibre Channel controller connectable to the storage unit (column 61, lines 55-60).

Instead, column 61, lines 55-60 of Muller states:

Reserving sixty-four bytes at the beginning of a buffer also allows header information to be modified or prepended if necessary. For example, a regular Ethernet header of a packet may, because of routing requirements, need to be replaced with a much larger FDDI (Fiber Distributed Data Interface) header. One skilled in the art will recognize the...

The above-quoted passage that was cited by the Office Action does not teach or suggest “the interface device of claim 1, further comprising a Fibre Channel controller connectable to the storage unit,” as recited in claim 4. For at least this reason, claim 4 is not anticipated by Muller.

Regarding claim 7, the Office Action states:

As per claim 7, Muller further discloses the interface device of claim 1, further comprising a file cache adapted to store said data under control of a file system in the host (column 56, lines 20-30, column 58, lines 26-30, column 61, lines 34-35, column 62, lines 47-52).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 7 and respectfully assert that none of those passages teaches or suggests “The network interface device of claim 1, further comprising a file cache adapted to store said data under control of a file system in the host.” For at least this reason, claim 7 is not anticipated by Muller.

Regarding claim 8, the Office Action states:

As per claim 8, Muller discloses an interface device, connectable to a network, to a storage unit and to a host having a CPU running a file system and a protocol processing stack, the interface device comprising:

- A memory having first and second portions, said first portion adapted to store a communication control block defining a network message that includes a plurality of packets containing data and control information, and said second portion adapted to cache file blocks that are stored on said storage unit, said communication control block indicating a source location and a destination location for said message such that one of said source and destination locations is disposed in said second portion of said memory, and the other of said source and destination locations is disposed on the network (column 4, lines 20-25, column 7, lines 20-45, column 9, lines 14-25, 56-58, column 10, lines 1-7, column 11, lines 46-59, column 12, lines 11-15, column 52, lines 64-67, column 53, lines 1-7);
- Circuitry adapted to categorize headers of said packets (column 4, lines 48-50, column 7, lines 20-25, 31-35, 64-67, column 8, lines 1-5, 17-20, 50-60, column 9, lines 1-5, column 15, lines 35-38, column 35, lines 53-61, column 36, lines 11-30);
- A processor adapted to associate said packets with said communication control block for sending said data to said destination without processing by the CPU (column 4, lines 45-50, column 8, lines 50-60, column 9, lines 13-17, 22-35, 66-67, column 10, lines 2-7, column 11, lines 46-59, column 12, lines 11-15).

Applicants respectfully disagree with several of the foregoing Office Action statements regarding claim 8. For instance, applicants respectfully assert that neither column 4, lines 20-25, column 7, lines 20-45, column 9, lines 14-25, 56-58, column 10, lines 1-7, column 11, lines 46-59, column 12, lines 11-15, column 52, lines 64-67, nor column 53, lines 1-7 of Muller discloses “A memory having first and second portions, said first portion adapted to store a communication control block defining a network message that includes a plurality of packets containing data and control information, and said second portion adapted to cache file blocks that are stored on said storage unit, said communication control block indicating a source location and a destination location for said message such that one of said source and destination locations is disposed in said second portion of said memory, and the other of said source and destination locations is disposed on the network.” Instead, column 4, lines 20-25 of Muller state:

In this embodiment of the invention a high performance network interface includes a flow database and a flow database manager module. A flow database in this embodiment contains an entry for each valid or active communication flow received by the network interface. Each flow may be identified by a flow key, stored in the flow's database entry, and may be indexed by a flow number.

Similarly, column 7, lines 20-25 of Muller state:

In embodiments of the invention described below, a NIC receives a packet from a network on behalf of a host computer system or other communication device. The NIC analyzes the packet (e.g., by retrieving certain fields from one or more of its protocol headers) and takes action to increase the efficiency with which the packet is transferred or provided to its destination entity. Equipment and methods discussed below for increasing the efficiency of processing or transferring packets received from a network may also be used for packets moving in the reverse direction (i.e., from the NIC to the network).

One technique that may be applied to incoming network traffic involves examining or parsing one or more headers of an incoming packet (e.g., headers for the layer two, three and four protocols) in order to identify the packet's source and destination entities and possibly retrieve certain other information. Using identifiers of the communicating entities as a key, data from multiple packets may be aggregated or re-assembled. Typically, a datagram sent to one destination entity from one source entity is transmitted via multiple packets. Aggregating data from multiple related packets (e.g., packets carrying data from the same datagram) thus allows a datagram to be re-assembled and collectively transferred to a host computer. The datagram may then be provided to the destination entity in a highly efficient manner. For example, rather than providing data from one packet ...

Column 9, lines 14-25 and 56-58 of Muller also do not teach what is alleged in the Office Action, but instead state:

108. The header parser may also provide the flow database manager with other information concerning the packet that was retrieved from the packet (e.g., length of the packet).

Flow database manager 108 searches FDB 110 in response to a query received from header parser 106. Illustratively, flow database 110 stores information concerning each valid communication flow involving a destination entity served by NIC 100. Thus, FDBM 108 updates FDB 110 as necessary, depending upon the information received from header parser 106. In addition, in this embodiment of the invention FDBM 108 associates an operation or action

...

Thus, after header parser 106 parses a packet FDBM 108 alters or updates FDB 110 and load distributor 112 identifies a processor in the host computer system to process the...

Similarly, column 10, lines 1-7 of Muller state:

Access) engine 120 and transfer to a host computer. In addition to storing the packet in a packet queue, a corresponding entry for the packet is made in control queue 118 and information concerning the packet's flow may also be passed to dynamic packet batching module 122. Control queue 118 contains related control information for each packet in packet queue 116.

Likewise, column 11, lines 46-59 of Muller state:

In state 136, information extracted from the packet's headers is forwarded to flow database manager 108 and/or load distributor 112. The FDBM uses the information to set up a flow in flow database 110 if one does not already exist for this communication flow. If an entry already exists for the packet's flow, it may be updated to reflect the receipt of a new flow packet. Further, FDBM 108 generates an operation code to summarize one or more characteristics or conditions of the packet. The operation code may be used by other modules of NIC 100 to handle the packet in an appropriate manner, as described in subsequent sections. The operation code is returned to the header parser, along with an index (e.g., a flow number) of the packet's flow in the flow database.

Continuing, column 12, lines 11-15 of Muller state:

Also in state 140, control information for the packet is stored in control queue 118 and information concerning the packet's flow (e.g., flow number, flow key) may be provided to dynamic packet batching module 122.

Finally, column 52, line 64 - column 53, line 7 of Muller state:

One Embodiment of a Control Queue

In one embodiment of the invention, control queue 118 stores control and status information concerning a packet received by NIC 100. In this embodiment, control queue 118 retains information used to enable the batch processing of protocol headers and/or the re-assembly of data from multiple related packets. Control queue 118 may also store information to be used by the host computer or a series of instructions operating on a host computer (e.g., a device driver for NIC 100). The information stored in control queue 118 may supplement or duplicate information stored in packet queue 116.

None of the above-quoted passages that were cited by the Office Action to reject claim 8 teaches or suggests "a memory having first and second portions, said first portion

adapted to store a communication control block defining a network message that includes a plurality of packets containing data and control information, and said second portion adapted to cache file blocks that are stored on said storage unit, said communication control block indicating a source location and a destination location for said message such that one of said source and destination locations is disposed in said second portion of said memory, and the other of said source and destination locations is disposed on the network,” as recited in claim 8. For at least this reason, claim 8 is not anticipated by Muller.

Regarding claim 9, the Office Action states:

As per claim 9, Muller discloses the interface device of claim 8, wherein said communication control block is created by the protocol-processing stack (column 6, lines 62-67, column 7, lines 57-63, column 8, lines 9-20, column 9, lines 43-47, column 10, lines 50-65).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 9 and respectfully assert that none of those passages teaches or suggests “The interface device of claim 8, wherein said communication control block is created by the protocol processing stack.” For at least this reason, claim 9 is not anticipated by Muller.

Regarding claim 10, the Office Action states:

As per claim 10, Muller discloses the interface device of claim 8, wherein said second portion of said memory is managed by the file system. (column 56, lines 20-30, column 58, lines 26-30, column 61, lines 34-35, column 62, lines 47-52).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 10 and respectfully assert that none of those passages teaches or suggests “The interface device of claim 8, wherein said second portion of said memory is managed by the file system.” For at least this reason, claim 10 is not anticipated by Muller.

Regarding claim 11, the Office Action states:

As per claim 11, Muller discloses the interface device of claim 8, wherein said circuitry is adapted to process a transport layer header of said headers. (column 4, lines 48-50, column 7, lines 20-25, 31-35, 64-67 column 8, lines 1-5, 17-20, 50-60, column 9, lines 1-5, column 15, lines 35-38, column 35, lines 53-67, column 36, lines 11-30).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 11 and respectfully assert that none of those passages teaches “The interface device of claim 8, wherein said circuitry is adapted to process a transport layer header of said headers.” For at least this reason, claim 11 is not anticipated by Muller.

Regarding claim 13, the Office Action states:

As per claim 13, Muller discloses the interface device of claim 8, wherein said processor is configured to associate at least one of said packets with said communication control block by creating a header for said packet that is based on said communication control block and adding said header to said packet. (column 9, lines 20-35, column 11, lines 46-60).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 13 and respectfully assert that none of those passages teaches “The interface device of claim 8, wherein said processor is configured to associate at least one of said packets with said communication control block by creating a header for said packet that is based on said communication control block and adding said header to said packet.” For at least this reason, claim 13 is not anticipated by Muller.

Regarding claim 15, the Office Action states:

As per claim 15, Muller discloses the interface device of claim 8, further comprising a network port adapted to connect with the storage unit. (column 4, lines 40-43, column 6, lines 37-40, column 7, lines 15-19, column 8, lines 40-43, column 9, lines 1-5, column 10, lines 65-67).

Applicants have reviewed the passages from Muller that the Office Action cites regarding claim 15 and respectfully assert that none of those passages teaches or suggests “The interface device of claim 8, further comprising a network port adapted to connect with the storage unit.” For at least this reason, claim 15 is not anticipated by Muller.

Regarding claim 16, the Office Action states:

As per claim 16, Muller discloses an interface device connectable to a local computer, a network and a storage unit, the local computer having a CPU and a protocol processing stack, the interface device comprising:

- A memory including a file cache for temporary storage of data being transferred between the network and the storage unit (column 56, lines 20-30, column 58, lines 26-30, column 61, lines 34-35, column 62, lines 47-52);

- Slow-path means for processing a first packet of a message by sending the packet to the local computer for processing by the CPU running the protocol stack (column 6, lines 62-67, column 7, lines 57-63, column 8, lines 9-20, column 9, lines 43-47, column 10, lines 50-65);
- Fast-path means for transferring a second packet of said message between the network and the storage unit without processing by the CPU (column 4, lines 45-50, 58-67 column 8, lines 50-60, 66-67, column 9, lines 13-17, 22-35, 66-67, column 10, lines 2-7, column 11, lines 46-59, column 12, lines 11-15, column 16, lines 59-67).

Applicants respectfully disagree with several of the foregoing Office Action statements regarding claim 16. For instance, applicants respectfully assert that neither column 56, lines 20-30, column 58, lines 26-30, column 61, lines 34-35, nor column 62, lines 47-52 of Muller discloses "A memory including a file cache for temporary storage of data being transferred between the network and the storage unit." Instead, column 56, lines 20-30 of Muller state:

Free ring manager 1012 attempts to ensure that a buffer is always available for a packet. Thus, in one embodiment of the invention free ring manager 1012 includes descriptor cache 1012a configured to store a number of descriptors (e.g., up to eight) at a time. Whenever there are less than a threshold number of entries in the cache (e.g., five), additional descriptors may be retrieved from the free descriptor ring. Advantageously, the descriptors are of such a size (e.g., sixteen bytes) that some multiple (e.g., four) of them can be efficiently retrieved in a sixty-four byte cache line transfer from the host computer.

Similarly, column 58, lines 26-30 of Muller state:

...or software operating on the host computer (e.g., a driver associated with NIC 100). In one embodiment of the invention, completion ring manager 1014 includes completion descriptor cache 1014a. Completion descriptor cache 1014a may store one or more completion descriptors for collective transfer from DMA engine 120 to the host computer.

Similarly, column 61, lines 34-35 of Muller state:

...are not split). In this alternative embodiment, a cache line of storage (e.g., sixty-four bytes for a SolarisTM workstation) is...

Similarly, column 62, lines 47-52 of Muller state:

...the free descriptor ring and stores them in a descriptor cache until a buffer is needed, at which time the buffer's buffer identifier is stored in the free buffer array. In yet another embodiment, a descriptor may be used

(e.g., the buffer that it references may be used to store a packet) while still in the cache.

None of the above-quoted passages that were cited by the Office Action to reject claim 16 teaches or suggests “a memory including a file cache for temporary storage of data being transferred between the network and the storage unit,” as recited in claim 16. For at least this reason, claim 16 is not anticipated by Muller.

II. 35 U.S.C. §103

A. The Standard

“The combination of elements from non-analogous sources, in a manner that reconstructs the applicant's invention only with the benefit of hindsight, is insufficient to present a prima facie case of obviousness. There must be some reason, suggestion, or motivation found in the prior art whereby a person of ordinary skill in the field of the invention would make the combination. That knowledge can not come from the applicant's invention itself.” *In re Oetiker*, 24 USPQ 2d 1443, 1446 (Fed. Cir. 1992). See also *In re Clay*, 966 F.2d 656, 658-660 (Fed. Cir. 1992) and *Monarch Knitting Mach. Corp. v. Sulzer Morat GmbH*, 139 F.3d 877, 881 (Fed. Cir., 1998). “Obviousness cannot be established by combining the teachings of the prior art to produce the claimed invention, absent some teaching or suggestion supporting the combination. Under section 103, teachings of references can be combined *only* if there is some suggestion or incentive to do so...The mere fact that the prior art may be modified in the manner suggested by the Examiner does not make the modification obvious unless the prior art suggested the desirability of the modification.” *In re Fritch*, 972 F.2d 1260, 1266 (Fed. Cir. 1992). “In proceedings before the Patent and Trademark Office, the Examiner bears the burden of establishing a prima facie case of obviousness based upon the prior art.” *Id.* at 1265.

B. The Office Action Assertions

Claims 2, 5 and 14 stand rejected under 35 U.S.C. §103(a). The Office Action states:

Claims 2, 5 and 14 stand rejected under 35 U.S.C. 103(a) as being unpatentable over Muller et al. (hereinafter "Muller", 6,650,640) in view of Day et al. (hereinafter "Day", 6,065,096).

Regarding claim 2, the Office Action states:

As per claim 2, Muller discloses the interface device of claim 1. Muller does not explicitly disclose the interface further comprising a SCSI controller connectable to the storage unit.

However, Day discloses SCSI interface channels attached to disk drives (column 2, lines 40-54, column 5, lines 1-25).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to implement or incorporate in Muller's device Day's interface comprising a SCSI controller in order to provide for a simple, lower cost RAID controller architecture to enable lower cost and complexity associated with high performance and high reliability storage subsystems.

As noted above regarding claim 1, Muller does not teach or suggest "a mechanism for associating said packet with said control information and for selecting whether to process said packet by said computer or to send data from said packet to the storage unit, thereby avoiding the computer." Day also does not teach or suggest this limitation. Implementing or incorporating in Muller's device Day's interface as proposed by the Office Action would not solve this deficiency.

Moreover, the motivation to make this modification is contradicted by the reasoning cited by the Office Action. Somehow implementing or incorporating in Muller's device Day's interface would increase the cost and complexity of Muller's device without any obvious benefit. Muller's device "relates to a network interface circuit (NIC) for processing communication packets exchanged between a computer network and a host computer system" (column 1, lines 51-54). Day's RAID controller, on the other hand, provides a chip for controlling a redundant array of inexpensive disk drives (column 1, lines 11-19; column 2, lines 11-24). Stated differently, Muller involves network communication and Day involves storage. Absent the teachings of the present invention, no motivation is apparent in the cited references to make the modification proposed by the Office Action.

Regarding claim 5, the Office Action states:

As per claim 5, Muller discloses the interface device of claim 1. Muller does not explicitly disclose the interface further comprising a RAID controller connectable to the storage unit.

However, Day discloses a RAID controller that integrates onto a single integrated circuit of a general purpose processor (column 2, lines 11-25, 55-67).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to implement or incorporate in Muller's device Day's interface comprising a RAID controller allowing the disk interface connections and protocols to be more flexibly selected but at the cost of less integration within the circuit.

As noted above regarding claim 1, Muller does not teach or suggest "a mechanism for associating said packet with said control information and for selecting whether to process said packet by said computer or to send data from said packet to the storage unit, thereby avoiding the computer." Day also does not teach or suggest this limitation. Implementing or incorporating in Muller's device Day's interface as proposed by the Office Action would not solve this deficiency.

Moreover, the motivation asserted by the Office Action to make this modification is quoted from Day at column 2, lines 51-54, which refers to an alternative storage configuration, not anything involving network communication. As mentioned above, somehow implementing or incorporating in Muller's device Day's interface would increase the cost and complexity of Muller's device without any obvious benefit. Muller's device "relates to a network interface circuit (NIC) for processing communication packets exchanged between a computer network and a host computer system" (column 1, lines 51-54). Day's RAID controller, on the other hand, provides a chip for controlling a redundant array of inexpensive disk drives (column 1, lines 11-19; column 2, lines 11-24). Stated differently, Muller involves network communication and Day involves storage. Absent the teachings of the present invention, no motivation is apparent in the cited references to make the modification proposed by the Office Action.

Regarding claim 14, the Office Action states:

As per claim 14, Muller discloses the interface device of claim 8. Muller does not explicitly disclose the interface further comprising a SCSI controller adapted to connect with the storage unit.

However, Day discloses SCSI interface channels attached to disk drives (column 2, lines 40-54, column 5, lines 1-25).

Therefore, one of ordinary skill in the art at the time the invention was made would have found it obvious to implement or incorporate in Muller's device Day's interface comprising a SCSI controller in order to provide for a simple, lower cost RAID controller architecture to enable lower cost and complexity associated with high performance and high reliability storage subsystems.

As noted above regarding claim 8, Muller does not teach or suggest "a memory having first and second portions, said first portion adapted to store a communication control block defining a network message that includes a plurality of packets containing data and control information, and said second portion adapted to cache file blocks that are stored on said storage unit, said communication control block indicating a source location and a destination location for said message such that one of said source and destination locations is disposed in said second portion of said memory, and the other of said source and destination locations is disposed on the network," as recited in claim 8. Day also does not teach or suggest this limitation. Implementing or incorporating in Muller's device Day's interface as proposed by the Office Action would therefore still have nonobvious differences from the invention recited in claim 14.

Moreover, the motivation to make this modification is contradicted by the reasoning cited by the Office Action. Somehow implementing or incorporating in Muller's device Day's interface would increase the cost and complexity of Muller's device without any apparent benefit. Muller's device "relates to a network interface circuit (NIC) for processing communication packets exchanged between a computer network and a host computer system" (column 1, lines 51-54). Day's RAID controller, on the other hand, provides a chip for controlling a redundant array of inexpensive disk drives (column 1, lines 11-19; column 2, lines 11-24). Stated differently, Muller involves network communication and Day involves storage. Absent the teachings of the present invention, no motivation is apparent in the cited references to make the modification proposed by the Office Action.

III. The Amendment to the Claims

Applicants have amended claims 1 and 16 to include the limitation "the storage unit including a disk drive" because applicants intend to focus their operating business on providing functionality to systems that include disk drives. Similarly, applicants have added new claim 19 to recite a comparable limitation. Support for these limitations can be found, for example, on page 4, lines 16 and 17, page 7, lines 26 and 30 and page 8, line 4.

Applicants have added new claims 17, 18 and 20 to recite iSCSI limitations. Support for these limitations can be found, for example, on page 26, lines 6 and 31, page 27, line 2 and page 29, line 17.

Applicants have added four new claims, bringing the total number of claims to twenty, so that no extra claims fee is due.

IV. Conclusion

In this Amendment, applicants have responded to each item of the Office Action by explaining why the Office Action does not present a prima facie case of anticipation or obviousness for any of the claims. As such, applicants respectfully assert that the application is in condition for allowance, and a notice of allowance is solicited.

Respectfully submitted,

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to the Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on March 11, 2005.

Date: 3-11-05



Mark Lauer



Mark Lauer
Reg. No. 36,578
6601 Koll Center Parkway
Suite 245
Pleasanton, CA 94566
Tel: (925) 484-9295
Fax: (925) 484-9291